

WEST Search History

DATE: Tuesday, July 01, 2003

Set Name Query

side by side

Hit Count Set Name

result set

DB=USPT; PLUR=YES; OP=ADJ

L13 5987249.pn. and stub

1 L13

L12 6332213.pn. and stub

1 L12

DB=JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=ADJ

L11 6332213.pn. and stub

0 L11

L10 L9 and compil\$

2 L10

L9 instrument\$ and stub

107 L9

DB=USPT,PGPB; PLUR=YES; OP=ADJ

L8 L7 and L1

27 L8

L7 (((717/124 |717/125 |717/126 |717/127 |717/128 |717/129 |717/130
|717/131 |717/132 |717/133)!.CCLS.))

1038 L7

DB=USPT; PLUR=YES; OP=ADJ

L6 L5 and GUI

29 L6

L5 L2 and compil\$

61 L5

L4 L2 and GUI

30 L4

L3 L2

108 L3

DB=USPT,PGPB; PLUR=YES; OP=ADJ

L2 L1 and pars\$

159 L2

L1 instrument\$ and stub

3561 L1

END OF SEARCH HISTORY

Searching for **stubs and instrumentation**.

Restrict to: [Header](#) [Title](#) Order by: [Citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Amazon](#) [B&N](#) [Google \(RI\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)

25 documents found. Order: citations weighted by year.

[The Coign Automatic Distributed Partitioning System - Hunt, Scott \(1999\) \(Correct\) \(10 citations\)](#)
 standard, COM can transparently interpose proxies, **stubs**, and middleware layers between communicating of the DCOM code, including interface proxies and **stubs**, within the application's address space. Coign At load time, the Coign runtime inserts binary **instrumentation** into the images of system libraries in the ftp.cs.rochester.edu/pub/papers/systems/99.OSDI.Coign_automatic_distributed_partitioning_system.pdf.gz

[On the Marginal Utility of Network Topology Measurements - Barford, Bestavros.. \(2001\) \(Correct\) \(2 citations\)](#)
 directed graph from a source is a tree. Leaf **Stub** Border Backbone Fig. 1. Classification of organizes nodes into one of four types: leafs, **stubs**, border and backbone illustrated in Figure 1. For algorithms for effective placement of Internet **instrumentation** in the context of their IDMaps project, a www.cs.bu.edu/faculty/crovella/paper-archive/imw-marginal-utility.pdf

[Intercepting and Instrumenting COMApplications - Galen Hunt Michael \(1999\) \(Correct\) \(5 citations\)](#)
 virtual function table. Interface Proxies and **Stubs** Location transparency is achieved through transparency is achieved through proxies and **stubs** generated by the MIDL compiler. Proxies marshal the implementation of an interception and **instrumentation** system tested on over 300 COM binary research.microsoft.com/~galenh/Publications/HuntCoots99.pdf

[Making Real-Time Reactive Systems Reliable - Marzullo, Wood \(1991\) \(Correct\) \(12 citations\)](#)
 predicate. Sensors and actuators are implemented by **stubs** that run on machines supporting the instrumented registers a procedure with the sensor **stub**. The **stub** is responsible for responding to poll for distributed application management are **instrumentation** of the application, representation of the <ftp.cs.ucsd.edu/pub/faculty/marzullo/TR90-1155.ps.Z>

[Ryo: A Versatile Instruction Instrumentation Tool For Pa-Risc - Zucker, Karp \(1995\) \(Correct\) \(3 citations\)](#)
 an unconditional jump to the RYOLS library calling **stub**. This **stub** saves the state of the machine, does jump to the RYOLS library calling **stub**. This **stub** saves the state of the machine, does any necessary Ryo: A Versatile Instruction **Instrumentation** Tool For Pa-Risc Daniel F. Zucker And Alan umunhum.stanford.edu/tr/zucker.jan95.tr658.ps.Z

[A Guided Tour of the Coign Automatic Distributed Partitioning.. - Hunt, Scott \(1998\) \(Correct\) \(1 citation\)](#)
 Dam, ICOPS pioneered the use of compiler-generated **stubs** for inter-process communication. ICOPS was the of the DCOM code, including the interface proxy and **stub**, within the application's address space. Coign utility to insert the Coign profiling 3 **instrumentation** package, see Figure 1. Setcoign makes two www.cs.cmu.edu/People/bumba/filing_cabinet/.papers/hunt-coign-tr2.ps.gz

[Tools for Constructing Distributed Reactive Systems - Marzullo, Wood \(1991\) \(Correct\) \(3 citations\)](#)
 and Service A context is implemented with a Meta **stub**. A Meta **stub** is analogous to an RPC server **stub** A context is implemented with a Meta **stub**. A Meta **stub** is analogous to an RPC server **stub** (e.g.4]and ftp.cs.ucsd.edu/pub/faculty/marzullo/TR91-1187.ps.Z

[Coign: Efficient Instrumentation for Inter-Component.. - Hunt, Scott \(1997\) \(Correct\) \(1 citation\)](#)
 for the programmer's language of choice and RPC **stubs** for marshaling and unmarshaling function measure internal communication by leveraging IDL **stubs** to quantify the amount of data that would have Coign: Efficient **Instrumentation** for Inter-Component Communication Analysis hypatia.dcs.qmw.ac.uk/data/edu/cs.rochester.edu/systems/97.tr648.Coign_for_inter-component_communication_analysis.ps.gz

[Selecting Test Cases for Object Oriented Programs - Liu \(1996\) \(Correct\) \(1 citation\)](#)
 for testability, and managing test scaffolding (**stubs**, drivers, **instrumentation** code) Different and managing test scaffolding (**stubs**, drivers, **instrumentation** code) Different testing takes place in the vlsi.uwaterloo.ca/~wblu/testinfo/pubs/proposal.ps

Performing Replay in an OSF DCE Environment - Yuh Ming (1995) (Correct) (1 citation)

are captured by code in the client and server **stubs**. We have endeavoured to capture events with as **instrumentation**. We originally modified the **stubs** by postprocessing the **stub** code produced by the effect entirely when using software-based **instrumentation**. We originally modified the **stubs** by ccnga.uwaterloo.ca/pub/papers/Ps/conf11.ps.Z

Designing, Modelling and Implementing a Toolkit for Aspect-oriented .. - Low (Correct)

implements the remote methods must now have their **stubs** built again, then the application can run be introduced in the System Under Test. This **instrumentation** can be done manually, but this "cut and concern" we used AspectJ to develop the **instrumentation** aid TAST for Java applications. TAST lg/www.epfl.ch/workshops/aosd-uml/Allsubs/low.pdf

A Comparison of System Monitoring Methods, Passive.. - Moore, McGregor, Breen (Correct)

Network Kernel **Instrumentation** System Call **Stubs** Block Cache Rpc Xdr Ip Device Driver Udp/tcp methods, passive network monitoring and kernel **instrumentation** A. W. Moore A. J. McGregor y J. W. passive network monitoring and kernel **instrumentation**. The comparison is made on the basis of www.cl.cam.ac.uk/~awm22/publication/paper3.ps.gz

Playing Inside the Black Box: Using Dynamic.. - Miller.. (Correct)

library replaces the standard system calls with RPC **stubs** that forward the calls back to the user's Playing Inside the Black Box: Using Dynamic **Instrumentation** to Create Security Holes Abstract Programs and control the running program. Dynamic **instrumentation** provides the necessary technology for this www.cs.wisc.edu/~kosart/dyn-security.pdf

An Approach to the Transparent Management.. - Villagr  . (2002) (Correct)

order to be managed. The use of class wrappers and **stub instrumentation** in object-based distributed that the EJB container generates all the necessary **stubs** and skeletons for RMI-IIOP communications as An Approach to the Transparent Management **Instrumentation** of Distributed Applications Vctor A. jungla.dit.upm.es/~jlopez/publicaciones/noms02.pdf

Using Shim Technology to Monitor DCE Runtime Performance - Oster, Bunt (1997) (Correct)

routines) would require editing the **stub** code generated by the IDL compiler 2 It may insert the necessary **instrumentation** code into the **stubs** automatically. While this approach may be of the software and hand-tuning of the **instrumentation** code in order to get the desired data. Since ftp.cs.usask.ca/pub/discus/paper.97-1.ps.Z

A Java-based Remote Laboratory for Distance Learning - Sam Hsu Bassem (2000) (Correct)

each created for a specific task on the server. The **stub** and skeleton are the mechanism used for invoking the different object sitting on the server. The **stubs** and skeletons of the interface are generated special local computer interface with a proper **instrumentation** device. This local computer is then setup as www.cse.fau.edu/~bassem/Publications/Pub-33-C-ICEE2000-Taiwan.PDF

AGuided Tour of the Coign Automatic Distributed Partitioning.. - Galen Hunt Michael (Correct)

Dam, ICOPS pioneered the use of compiler-generated **stubs** for inter-process communication. ICOPS was the of the DCOM code, including the interface proxy and **stub**, within the application's address space. Coign setcoign utility to insert the Coign profiling **instrumentation** package, see Figure 1. Setcoign makes two research.microsoft.com/~galenh/Publications/HuntEdoc98.pdf

Process Hijacking - Ki Ng (Correct)

by the shadow via remote procedure call. The RPC **stubs** for the application are contained in a library hijacker rewrites the application to use the RPC **stubs** instead of the standard system calls. The most by DynInst in the mutatee, called the run-time **instrumentation** library RTinst) DynInst can operate on www.cs.wisc.edu/paradyn/papers/hijack.ps.gz

The Trials and Tribulations of Building an Adaptive User.. - Korvemaker, Greiner (1999) (Correct)

Davison/Hirsh (1998) work (for learning "command **stubs**" we then explore several ways of extending and and Hirsh, DH97 DH98) which predicts user command **stubs** (commands without options and parameters, e.g. event. Of course, this may require elaborate **instrumentation**. Figure 3: Display of the Implementation www.cs.ualberta.ca/~greiner/PAPERS/AUI.ps

First 20 documents [Next 20](#)


Try your query at: [Amazon](#) [Barnes & Noble](#) [Google \(RI\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)

CiteSeer - citeseer.org - [Terms of Service](#) - [Privacy Policy](#) - Copyright © 1997-2002 [NEC Research Institute](#)

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

- ☐ By Author
- ☐ Basic
- ☐ Advanced

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library
-  Print Format

Your search matched **11** of **945031** documents.

A maximum of **11** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.

You may refine your search by editing the current search expression or entering a new one in the text box.

Then click **Search Again**.

(Instrumentation)and (stub)

Search Again

Results:

Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 The ICRF computer control and monitoring system

Mervine, J.; Sichta, P.; Wilson, R.; Giles, D.; Skelly, G.;

Fusion Engineering, 1989. Proceedings., IEEE Thirteenth Symposium on , 2-6 Oct. 1989

Page(s): 455 -458 vol.1

[\[Abstract\]](#) [\[PDF Full-Text \(184 KB\)\]](#) **IEEE CNF**

2 Modeling, simulation, and measurement considerations of high-speed digital buses

Novak, I.;

Instrumentation and Measurement Technology Conference, 1992.

IMTC '92., 9th IEEE , 12-14 May 1992

Page(s): 147 -151

[\[Abstract\]](#) [\[PDF Full-Text \(372 KB\)\]](#) **IEEE CNF**

3 System control and data acquisition of the two new FWCD RF systems at DIII-D

Harris, T.E.; Allen, J.C.; Cary, W.P.; Ferguson, S.W.; Petty, C.C.;

Pinsker, R.I.;

Fusion Engineering, 1995. SOFE '95. 'Seeking a New Energy Era',

16th IEEE/NPSS Symposium , Volume: 2 , 30 Sept.-5 Oct. 1995

Page(s): 878 -880 vol.2

[\[Abstract\]](#) [\[PDF Full-Text \(320 KB\)\]](#) **IEEE CNF**

4 Coupling cavity damper for the ARES

Naito, F.; Akai, K.; Akasaka, N.; Ezura, E.; Kageyama, T.; Mizuno, H.; Nakanishi, H.; Takeuchi, Y.; Yamazaki, Y.; Kobayashi, T.;

Particle Accelerator Conference, 1997. Proceedings of the 1997 ,
Volume: 3 , 12-16 May 1997

Page(s): 2977 -2979 vol.3

[\[Abstract\]](#) [\[PDF Full-Text \(244 KB\)\]](#) **IEEE CNF**

5 A design pattern based approach to generating synchronization adaptors from annotated IDL

Jacobsen, H.-A.; Kramer, B.J.;

Automated Software Engineering, 1998. Proceedings. 13th IEEE
International Conference on , 13-16 Oct. 1998

Page(s): 63 -72

[\[Abstract\]](#) [\[PDF Full-Text \(236 KB\)\]](#) **IEEE CNF**

6 The level-2 muon trigger at D0

Fortner, M.; Maciel, A.; Evans, H.; Kothari, B.; Uzunyan, S.;

Nuclear Science Symposium Conference Record, 2000 IEEE , Volume:
2 , 15-20 Oct. 2000

Page(s): 12/6 -12/9 vol.2

[\[Abstract\]](#) [\[PDF Full-Text \(344 KB\)\]](#) **IEEE CNF**

7 Reduction and characterization of a drift error in measurement of small antennas using a network analyzer

Ida, I.; Sato, J.; Sekizawa, T.; Yoshimura, H.; Ito, K.;

Antennas, Propagation and EM Theory, 2000. Proceedings. ISAPE
2000. 5th International Symposium on , 15-18 Aug. 2000

Page(s): 504 -507

[\[Abstract\]](#) [\[PDF Full-Text \(288 KB\)\]](#) **IEEE CNF**

8 Measurement and simulation of crosstalk reduction by discrete discontinuities along coupled PCB traces

Novak, I.; Eged, B.; Hatvani, L.;

Instrumentation and Measurement, IEEE Transactions on , Volume:
43 Issue: 2 , Apr 1994

Page(s): 170 -175

[\[Abstract\]](#) [\[PDF Full-Text \(400 KB\)\]](#) **IEEE JNL**

9 Modeling, simulation, and measurement considerations of high-speed digital buses

Novak, I.;

Instrumentation and Measurement, IEEE Transactions on , Volume: 41 Issue: 6 , Dec. 1992

Page(s): 921 -925

[\[Abstract\]](#) [\[PDF Full-Text \(376 KB\)\]](#) **IEEE JNL**

10 Noise measurements of microwave transistors using an uncalibrated mechanical stub tuner and a built-in reverse six-port reflectometer

Di-Luan Le; Ghannouchi, F.M.;

Instrumentation and Measurement, IEEE Transactions on , Volume: 44 Issue: 4 , Aug. 1995

Page(s): 847 -852

[\[Abstract\]](#) [\[PDF Full-Text \(396 KB\)\]](#) **IEEE JNL**

11 The Level-2 muon trigger at D0

Fortner, M.; Maciel, A.; Evans, H.; Kothari, B.; Uzunyan, S.;

Nuclear Science, IEEE Transactions on , Volume: 49 Issue: 4 , Aug. 2002

Page(s): 1589 -1592

[\[Abstract\]](#) [\[PDF Full-Text \(197 KB\)\]](#) **IEEE JNL**

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#)
[Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#)
[No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Copyright © 2003 IEEE — All rights reserved



Subscribe Register
(Full Service) (Limited Service, Free)

Login

Search: ☒ The Guide ☒ The ACM Digital Library

+instrumentation +stub +class +parsing

THE ACM DIGITAL LIBRARY

Feedback

Terms used instrumentation stub class parsing

Sort results
by

relevance



Display results

expanded form



Save results to a Binder

Search Tips

☐ Open results in a new window

Results 1 - 20 of 26

Result page: **1** 2 r

1 A tool framework for static and dynamic analysis of object-oriented softw

Kathleen A. Lindlan, Janice Cuny, Allen D. Malony, Sameer Shende, Forschungs Rasmussen, Bernd Mohr

November 2000

Proceedings of the 2000 ACM/IEEE conference on Supercom

Full text available: pdf(2.67 MB)

Additional Information: full citation, abstract, refer

The developers of high-performance scientific applications often work in comp demands on program analysis tools. The developers need tools that interoper and provide source-level feedback. In this paper, we describe a tool framewo supports the development of program analysis tools meeting these requireme create a comp ...

2 Middleware performance analysis: Performance monitoring of java applic

M. Harkema, D. Quartel, B. M. M. Gijzen, R. D. van der Mei

July 2002





Proceedings of the third international workshop on Software and p

Full text available: pdf(219.69 KB)

Additional Information: full citation, abstract, ref

Over the past few years, Java has evolved into a mature platform for develop the commercial success of these applications is end-to-end performance, e.g. availability. This raises the need for the development, validation and analysis metrics of interest. To develop and validate performance models, insight in th essent ...

Keywords: performance measurement and monitoring of java applications

- 3 Automatic performance prediction to support cross development of paral**
Matthias Schumann
January 1996 Proceedings of the SIGMETRICS symposium on Parallel and distrib
Full text available:  pdf(1.32 MB) Additional Information: full citation, references, index terms
- 4 Technical papers: dynamic program analysis: Tracking down software bu**
Sudheendra Hangal, Monica S. Lam
May 2002 Proceedings of the 24th international conference on Software eng
Full text available:  pdf(1.30 MB) Additional Information: full citation, abstract, re
This paper introduces DIDUCE, a practical and effective tool that aids program
identifying their root causes. By instrumenting a program and observing its b
formulates hypotheses of invariants obeyed by the program. DIDUCE hypothe
and gradually relaxes the hypothesis as violations are detected to allow for ne
to catch soft ...
- 5 Parse tree annotations**
James J. Purtilo, John R. Callahan
December 1989 Communications of the ACM, Volume 32 Issue 12
Full text available:  pdf(1.09 MB) Additional Information: full citation, abstract, references, ci
A technique for associating rewrite rules with productions so that many high-l
generated easily is described. While eclipsed in power by other editing and co
management of both synthesized and inherited attributes, this approach is es
power to deal with a wide class of problems arising from practical applications
Keywords: Attribute grammars, parsers, software interconnection systems, s
- 6 The architecture of Montana: an open and extensible programming enviro**
Michael Karasick
November 1998 ACM SIGSOFT Software Engineering Notes , Proceedings of the 6
Foundations of software engineering, Volume 23 Issue 6
Full text available:  pdf(1.16 MB) Additional Information: full citation, abstract, referenc
Montana is an open, extensible integrated programming environment for C++
linking, a persistent code cache called a CodeStore, and a set of programming
CodeStore serves as a central source of information for compiling, browsing,
about both the static and dynamic structure of the compiled program. This inf
function bodies, ...
Keywords: C++, compilation, extensible systems, frameworks, incremental co
environments, programming environments

- 7 Requirements for and evaluation of RMI protocols for scientific computing**
Madhusudhan Govindaraju, Aleksander Slominski, Venkatesh Choppella, Randal
November 2000 Proceedings of the 2000 ACM/IEEE conference on Supercom

Full text available:  pdf(306.83 KB)

Additional Information: full citation, abstract, ref


Distributed software component architectures provide a promising approach to Grid applications. Communication in these component architectures is based on protocols that allow one software component to invoke the functionality of another. Examples include the existing Java RMI and the new Simple Object Access Protocol (SOAP). SOAP has the advantages and component ...

8 Software reuse

Charles W. Krueger

June 1992

ACM Computing Surveys (CSUR), Volume 24 Issue 2

Full text available:  pdf(4.96 MB)

Additional Information: full citation, abstract, reference

Software reuse is the process of creating software systems from existing software instead of from scratch. This simple yet powerful vision was introduced in 1968. Software reuse is a standard software engineering practice. In an attempt to understand why, researchers have studied software reuse and in the obstacles to implementing it. This paper surveys the state of the art in the ...


Keywords: abstraction, cognitive distance, software reuse

9 Just-in-time aspects: efficient dynamic weaving for Java

Andrei Popovici, Gustavo Alonso, Thomas Gross

March 2003

Proceedings of the 2nd international conference on Aspect-oriented programming

Full text available:  pdf(1.15 MB)

Additional Information: full citation, abstract, reference

Recent developments in service architectures suggest that run-time adaptation is necessary. In this paper we discuss application requirements on run-time AOP support and how to meet them. We provide basic support for weaving using the Just-In-Time compiler and an interchangeable module on top of the basic support. This approach allows us to achieve flexibility, and secure we ...

10 SAFKASI: a security mechanism for language-based systems

Dan S. Wallach, Andrew W. Appel, Edward W. Felten

October 2000

ACM Transactions on Software Engineering and Methodology (TOSEM)

Full text available:  pdf(234.89 KB)

Additional Information: full citation, abstract, reference


In order to run untrusted code in the same process as trusted code, there must be a way to determine if their caller is authorized to exercise the privilege of using the dangerous technique called stack inspection to address this concern. But its original definition has an unclear relationship to the actual achievement of security, overconstrained by the need for performance.

Keywords: Internet, Java, WWW, access control, applets, security-passing style

11 Flick: a flexible, optimizing IDL compiler

Eric Eide, Kevin Frei, Bryan Ford, Jay Lepreau, Gary Lindstrom

May 1997 ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1997 conference implementation, Volume 32 Issue 5

Full text available:  pdf(1.75 MB)

Additional Information: full citation, abstract, citir

An interface definition language (IDL) is a nontraditional language for describing IDL compilers generate "stubs" that provide separate communicating process invocation or procedure call. High-quality stub generation is essential for application designs, whether the components reside on a single computer or on multiple

12 Automated Testing of Classes

Ugo Buy, Alessandro Orso, Mauro Pezze

August 2000 ACM SIGSOFT Software Engineering Notes , Proceedings of the International Analysis, Volume 25 Issue 5

Full text available:  pdf(396.92 KB)

Additional Information: full citation, abstract, re

Programs developed with object technologies have unique features that often Consider, for instance, the dependence between the state of an object and the method executed by an object often depends on the state of the object when that techniques for testing of classes exercise class methods when the metho

Keywords: class testing, data flow analysis, symbolic execution, testing and a

13 On objects and events

Patrick Th. Eugster, Rachid Guerraoui, Christian Heide Damm

October 2001 ACM SIGPLAN Notices , Proceedings of the OOPSLA '01 conference Languages and Applications, Volume 36 Issue 11

Full text available:  pdf(308.58 KB)


Additional Information: full citation, abstract, re

This paper presents linguistic primitives for publish/subscribe programming u primitives into a strongly typed object-oriented language through four mechn (3)closures, and (4) deferred code evaluation. We illustrate our primitives thr respective lacks. A precompiler transforms statements based on our publish/s generated ...

14 Type feedback vs. concrete type inference: a comparison of optimization

Ole Agesen, Urs Hölzle

October 1995 ACM SIGPLAN Notices , Proceedings of the tenth annual conference languages, and applications, Volume 30 Issue 10

Full text available:  pdf(2.27 MB)

Additional Information: full citation, abstract, referenc

Two promising optimization techniques for object-oriented languages are type prediction) and concrete type inference (static analysis). We directly compare effectiveness on a suite of 23 SELF programs while keeping other factors cons over 95% of all sends and deliver similar overall performance with one except integer ...

15 Human-computer interface development: concepts and systems for its m

H. Rex Hartson, Deborah Hix

March 1989

ACM Computing Surveys (CSUR), Volume 21 Issue 1

Full text available:  pdf(7.97 MB)

Additional Information: full citation, abstract, references, ci

Human-computer interface management, from a computer science viewpoint, human-computer interfaces, including their representation, design, implemen
This survey presents important concepts of interface management: dialogue i
representation, interactive tools, rapid prototyping, development methodologi
independence is th ...

16 Reconciling responsiveness with performance in pure object-oriented lar

Urs Hölzle, David Ungar

July 1996

ACM Transactions on Programming Languages and Systems (TOPLA

Full text available:  pdf(537.19 KB)

Additional Information: full citation, abstract, references,

Dynamically dispatched calls often limit the performance of object-oriented pr
encourages factoring code into small, reusable units, thereby increasing the fi
Frequent calls not only slow down execution with the dispatch overhead per s
optimization by limiting the range and effectiveness of standard global optimi:
calles prevent stand ...

Keywords: adaptive optimization, pause clustering, profile-based optimization

17 The design and performance of a pluggable protocols framework for real middleware

Carlos O'Ryan, Fred Kuhns, Douglas C. Schmidt, Ossama Othman, Jeff Parsons

April 2000

IFIP/ACM International Conference on Distributed systems platfo

Full text available:  pdf(231.64 KB)


Additional Information: full citation, abstract, re

*To be an effective platform for performance-sensitive real-tim
off-the-shelf CORBA middleware must preserve the communi
properties of applications end-to-end. However, the standard
protocols are not well suited for applications that cannot toler
latency, and jitter associated with general-purpose messaging
essential, therefore, to de ...*

18 Workshop on OODB implementation

Satish Thatte

January 1987

ACM SIGPLAN Notices , Addendum to the proceedings on Object-or
applications (Addendum), Volume 23 Issue 5Full text available:  pdf(1.49 MB)

Additional Information: full citation, citing

19 Towards automatic construction of staged compilers

Matthai Philipose, Craig Chambers, Susan J. Eggers

January 2002 ACM SIGPLAN Notices , Proceedings of the 29th ACM SIGPLAN-SIG languages, Volume 37 Issue 1

Full text available:  pdf(269.51 KB)


Additional Information: full citation, abstract,

Some compilation systems, such as offline partial evaluators and selective dyi optimizations. A staged optimization is one where a logically single optimizati stage(s) performing preplanning set-up work, given any available partial know the final stage completing the optimization. The final stage can be much faster of its work ...

20 Information-flow and data-flow analysis of while-programs

Jean-Francois Bergeretti, Bernard A. Carré

January 1985 ACM Transactions on Programming Languages and Systems (TO

Full text available:  pdf(1.57 MB)

Additional Information: full citation, abstract, references, c

Until recently, information-flow analysis has been used primarily to verify that variables cannot violate security requirements. Here, the notion of information development and validation. Information-flow relations are presented for while statements whose execution may cause information to be transmitted from or It ...

Results 1 - 20 of 26

Result page: **1** 2 next

The ACM Portal is published by the Association for Computing Machin

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#)

Useful downloads:  Adobe Acrobat  QuickTime  Windows



Subscribe Register
(Full Service) (Limited Service, Free)

Login

Search: ☒ The Guide ☒ The ACM Digital Library

+instrumentation +stub +class +parsing

THE ACM DIGITAL LIBRARY

Fee

Terms used instrumentation stub class parsing

Sort results
by

relevance



☒ Save results to a Binder

☒ Search Tips

☐ Open results in a new window

Display results

expanded form



Results 21 - 26 of 26

Result page: previous 1

21 TIGRA — an architectural style for enterprise application integrati

Wolfgang Emmerich, Ernst Ellmer, Henry Fieglein

July 2001

Proceedings of the 23rd international conference on Software engi

Full text available: pdf(137.99 KB)

Additional Information: full citation, abstract, referer

We report on experience that we made in the Trading room I (TIGRA) at a large German bank. TIGRA developed a distribu integrating different financial front-office trading systems wit applications. We generalize the experience by proposing an a for similar enterprise application integration tasks. The TIGRA/ data representation using domain-s ...

22 Evolving RPC for active storage

Muthian Sivathanu, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau

October 2002 Tenth international conference on architectural support for program

Proceedings of the 10th international conference on architectural su
operating systems (ASPLOS-X), Volume 37 , 36 , 30 Issue 10 , 5 ,

Full text available: pdf(1.56 MB)


Additional Information: full citation, abstr

We introduce Scriptable RPC (SRPC), an RPC-based framework that enables d active components. Technology trends point to a world where each componer or memory) has substantial computational capabilities; however, traditional n designed to take advantage of these new architectures, mandating wholesale powerful hardw ...

23 Data access for the masses through OLE DB

José A. Blakeley

June 1996 ACM SIGMOD Record , Proceedings of the 1996 ACM SIGMOD international conference on Database Management
Volume 25 Issue 2

Full text available:  pdf(1.24 MB)


Additional Information: full citation, abstract, references

This paper presents an overview of OLE DB, a set of interfaces being developed to allow applications to have uniform access to data stored in DBMS and non-DBMS in order to take advantage of the benefits of database technology without having to resort to proprietary interfaces. Our approach consists of defining an open, extensible Collection of interfaces that can be used as reusable portions ...

24 Process migration

September 2000

ACM Computing Surveys (CSUR), Volume 32 Issue 3

Full text available:  pdf(1.24 MB)

Additional Information: full citation, abstract, references


Process migration is the act of transferring a process between two machines. It is motivated by the need for resilience, eased system administration, and data access locality. Despite the benefits, process migration has not achieved widespread use. With the increasing deployment of distributed operating systems in particular, process migration is again receiving attention. As hardware and software development. As hardware and software development.

Keywords: distributed operating systems, distributed systems, load distribution, process migration

25 An object/message model for the development of integrated workstation

Kathleen Taylor, Michael A. Bauer

February 1990 Proceedings of the 1990 ACM SIGSMALL/PC symposium on Small Systems

Full text available:  pdf(1.26 MB)


Additional Information: full citation, abstract, references

There is a need for expandable, integrated workstation environments which can be modified to accommodate a particular user's needs. In this paper, a model for developing integrated workstation software. Integration is achieved through message passing between objects. The applications communicate with a mediator rather than each other.

26 DistView: support for building efficient collaborative applications using remote objects

Atul Prakash, Hyong Sop Shim

October 1994 Proceedings of the 1994 ACM conference on Computer supported Cooperative Work

Full text available:  pdf(1.61 MB)

Additional Information: full citation, abstract, references

The ability to share synchronized views of interactions with an application is crucial for collaborative work. This paper suggests a simple synchronous collaboration paradigm in which the interactions occur at the window level within a multi-user, multi-window application toolkit, DistView, that allows some of the application windows to be shared at the window level.

Keywords: active objects, collaboration technology, concurrency control, distributed interfaces, replicated objects, shared windows

Results 21 - 26 of 26

Result page: previous 1 **2**

The ACM Portal is published by the Association for Computing Machin

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#)

Useful downloads:  Adobe Acrobat  QuickTime  Windows